提取公共部分

1. 样式文件夹下面可以将各个功能的样式分而置之，相互引用，模块化管理。

脚手架使用

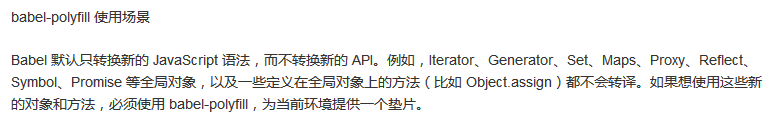
1. webpack可以在配置文件里面配置路径别名，stylus引用别名的时候前面加”~”.
2. Eslintrc文件下面rules里可以配置代码检测风格的配置

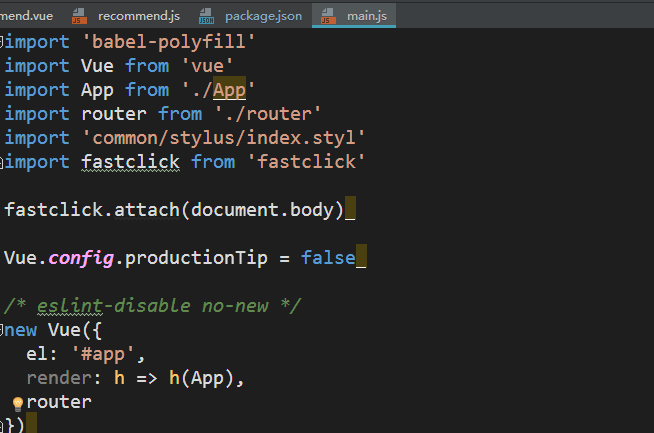
易错点

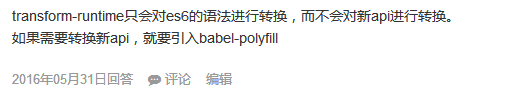
1. 注册组件是components 不是 component.
2. Props是一个对象，不是一个数组

Vue开发

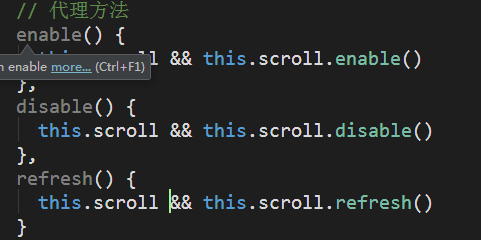
1. router被激活（当前router）时 ，会给这个<router-link>添加一个router-link-active的样式。
2. 给轮播图做初始化设置的时机:mounted()钩子里做，dom挂载完成之后，为了保证dom完全渲染完 可以加个setTimeout 20ms 因为浏览器17ms刷新一次。 详见，slider组件开发。
3. 开发过程中除了个问题：给轮播图初始化没有效果。原因是初始化时机不对导致，在mounted钩子里面初始化是元素挂载完成，但是请求数据是一个异步过程，也就是说元素挂载完成的时候轮播图的数据还没有生成，所以要保证轮播图有数据了才开始渲染slider 就可以加个if 条件是 数据有长度，这样既可以保证有数据的时候才开始渲染。 详见，slider组件开发
4. 组件里面要访问props的数据 可以通过this. 来访问
5. 生命钩子 destroyed 组件被销毁的时候 ----- 在这个钩子里面可以做一些操作，比如清除定时器这种，是一个好的习惯。
6. 给dom渲染绑定数据的时候，可以直接用’{{}}’，也可以用v-html绑定，有些数据里面需要html处理的数据就可以用v-html绑定
7. Babel-polyfill main.js中引入了这个插件，可以使用一些es6的api。





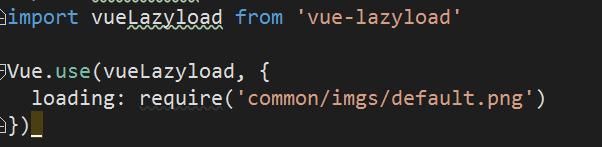


1. 方法代理：



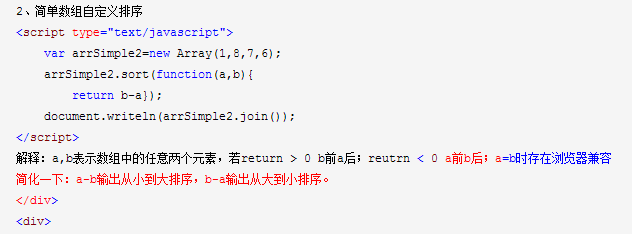
这是一个scroll组件开发 做了一个方法代理 可以让这个组件外部也能通过这个方法来操作所代理的方法（这里是指一些scroll的方法） 首先要this.scroll成立即为有这个对象，然后才能调用这个方法，这种思想真是太棒了 ----详见4-10 《scroll组件开发（上）》

1. vue-lazyload的问题，引入图片的时候用require引入loading图片，会被转换为base64编码，否则不生效-----此处不太理解 ---详见 《4-12懒加载插件使用》



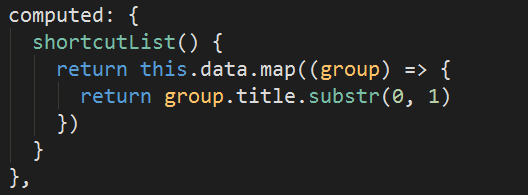
1504103834(1)

1. Better-scroll 和 fastclick 会有click冲突导致不能点击 给需要点击的目标加一个needsclick的类就可以解决了（这里我没有出现不能点击的情况）----详见 《4-12懒加载插件使用》
2. 数组的sort方法和根据首字母排序：



字母排序可以根据他的字符编码排序 使用charCodeAt()

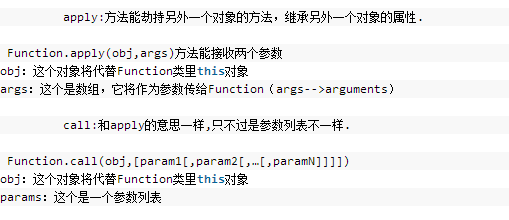
1. 数组的map方法：array.map(function(){}) 传入的函数可以返回一个新的数组，并用以前的数组内容作为基础生成新的数组

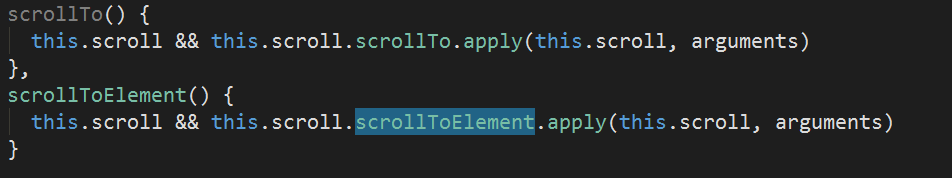


1. touch 事件的 touches： event的touches里面有一个数组，包含了关于这个事件的一些信息 所以要取里面的值 应该 e.touches[0]。
2. call 和 apply

两个的第一个参数都是调用方法的对象，最前面要跟上一个需要调用的函数，后面的参数为调用函数需要的参数 call是挨个写参数， apply是传入一个数组

本案例里面因为不确定需要传入哪些参数，所有用apply传入数组，这里的arguments就是值传进来的参数，即实参集合，也就是scollToElement的参数列表





Vue开发遗留的问题

1.axios的使用